CLAIMS

What is claimed is:

1. A processor comprising:
   an execution unit to execute a set of instructions; and
   an instruction fetching mechanism that retrieves the set of instructions to be executed by the execution unit, at least one of the set of instructions comprising a single instruction that provides for execution of other instructions of the set of instructions in accordance with multiple looping constructs.

2. The processor of claim 1, wherein the single instruction initializes a plurality of loops for later execution.

3. The processor of claim 1, wherein the multiple looping constructs are implemented in a nested structure.

4. The processor of claim 1, wherein the single instruction includes a plurality of loop termination conditions.

5. The processor of claim 1, wherein the single instruction includes at least one field that identifies a location of a last instruction of a loop.

6. A method of performing an instruction for use by a processor, the method comprising:

fetching the instruction from a memory source;

decoding the instruction to identify an instruction type; and

initializing a plurality of dedicated loop storage elements corresponding to a plurality of different loops to be executed using a single instruction.

7. The method of claim 6, wherein the storage elements include at least one of the following: an end-of-loop indicator, a start of loop indicator, a loop count, a loop register, and a condition code selection.

8. The method of claim 6, wherein the storage elements include a loop address for a start of a loop.

9. A method of performing an instruction for use by a processor, the method comprising:

fetching the instruction from a memory source;

decoding the instruction to identify an instruction type; and

determining a loop type for a single instruction that is to execute a plurality of different loops, the loop type comprising one of a conditional and non-conditional type of loop termination.

10. A method of executing at least one instruction by a processor, the method comprising:

fetching a first instruction from a memory source;

decoding the first instruction to identify an instruction type; and

determining a loop type selected from one of a conditional and non-conditional type of loop termination for a loop that contains more than one instruction other than the first instruction.

11. A method of executing at least one instruction by a processor, the method comprising:

fetching a first instruction from a memory source;

decoding the first instruction to identify an instruction type; and

determining a loop type selected from one of a conditional and non-conditional type of loop termination for a loop that contains at least one instruction that may be interrupted during loop execution.

12. A method of executing at least one instruction by a processor, the method comprising:

fetching a first instruction from a memory source;

decoding the first instruction to identify an instruction type;

performing a first set of logic relating to a first loop termination condition for a first loop; and

performing a second set of logic relating to a second loop termination condition for a second loop.

13. The method of claim 12, further comprising the step of determining whether the first loop termination condition is conditional or fixed and determining whether the second loop termination condition is conditional or fixed.

14. A method of executing instructions by a processor, the method comprising:
executing a first loop including a first instruction within a body of the first loop and a second loop including a first instruction within a body of the second loop, wherein the first instruction of the first loop and the first instruction of the second loop are a same instruction at the same address.

15. A method of executing instructions by a processor, the method comprising:
executing a first loop including a last instruction within a body of the first loop and a second loop including a last instruction within a body of the second loop, wherein the last instruction of the first loop and the last instruction of the second loop are the same instruction at the same address.

16. A processor instruction comprising:
a first field that indicates a first termination condition for a first execution loop; and
a second field that indicates a second termination condition for a second execution loop.

17. The processor instruction of claim 16, wherein the first execution loop is a same loop as the second execution loop.

18. The processor instruction of claim 16, further comprising a third field that indicates a first end-of-loop location.

19. The processor instruction of claim 18, further comprising a fourth field that indicates a second end-of-loop location.